

**Remarks/Arguments:**

Claims 1-20 are pending and stand rejected.

By this Amendment, claims 1, 3-5, 8, 16 and 19-20 are amended and claims 12 and 14-15 are canceled without prejudice.

No new matter is presented by the claim amendments. Support for the claim amendments can be found throughout the original specification and, for example, in the original specification at page 8, line 14 to page 9, line 24.

**Rejection of Claims 1, 8-9 and 13-15 under 35 U.S.C. §102(b)**

In the Office Action, at item 2, claims 1, 8-9 and 13-15 are rejected under 35 U.S.C. §102(b) as being anticipated by Kato (U.S. Patent Publication No. 2002/0051971).

Reconsideration is respectfully requested.

Claim 1 is directed to a method of task management, and recites:

assigning protection attributes indicating a memory range of a shared memory for each respective atomic sub-task of the one or more atomic sub-tasks such that each respective sub-task is executed by one of a plurality of processors which inherits access rights to the shared memory indicated by the protection attributes corresponding to the respective atomic sub-task.

Kato discloses that a processing task is broken down into a plurality of self-contained objects. Each of the task objects is defined with a computational task and at least one data-waiting slot for reception of data requested from another task object to which the processing task passes a message for the requested task. In Kato, parallel object task (POT) Engines are used to handle the parallel processing of tasks. (See Kato at paragraph [0046].) Kato further discloses that it is difficult to determine the size of the POT Objects which exist within the POT Engine until run time. Because of it's architecture, the POT Engines allocates a large number of POT Objects internally which changes dynamically. This requires a dynamic allocation of memory and in actuality uses, the system will have to handle a large number of POT Objects of differing sizes. Therefore, having a fast, efficient memory management is an extremely important issue. Therefore, the POT Engine is designed to contain its own memory management function, which realizes a faster performance than the default memory management implemented in the operating system. This memory management system is built to be thread-safe, in order to be able to deal with parallel processing correctly. (See Kato at paragraph [0065].) That is, Kato discloses the use of slots as a structure for passing messages between tasks and further discloses that the POT Engine includes its own memory management functionality. Kato, however, is silent regarding protection attributes and, more particularly,

"assigning protection attributes indicating a memory range of a shared memory for each respective atomic sub-task of the one or more atomic sub-tasks such that each respective sub-task is executed by one of the plurality of processors which inherits access rights to the shared memory indicated by the protection attributes corresponding to the respective atomic sub-task," as required by claim 1.

Accordingly, it is submitted that claim 1 patentably distinguishes over Kato for at least the above-mentioned reasons.

**Claim 8**

Claim 8, which includes similar but not identical features to those of claim 1, is submitted to patentably distinguish over Kato for at least similar reasons to those of claim 1.

**Claims 9 and 13-15**

Claims 9 and 13, which includes all of the limitations of claim 8, are submitted to patentably distinguish over Kato for at least the same reasons as claim 8.

Claims 14 and 15 have been canceled without prejudice. Accordingly, the rejection of these claims is now moot.

**Rejections of Claims 2-7, 10-12 and 16-20 under 35 U.S.C. §103(a)**

In the Office Action, at item 7, claims 2-7, 10-12 and 16-20 are rejected under 35 U.S.C. §103(a) as being unpatentable over Kato in view of Koning (U.S. Patent Publication No. 2002/0133530).

Reconsideration is respectfully requested.

**Claim 16**

Claim 16, which includes similar but not identical features to those of claim 1, is submitted to patentably distinguish over Kato for similar reasons to those of claim 1.

**Claims 2-7, 10-11 and 17-20**

Claims 2-7, 10-11 and 17-20, which include all of the limitations of claim 1, claims 8 or claim 16, are submitted to patentably distinguish over Kato for at least the same reasons as their respective independent claims.

**Claim 12**

Claim 12 has been canceled without prejudice. Accordingly, the rejection of claim 12 is now moot.

The addition of Koning does not overcome the deficiencies of Kato. This is because, Koning does not disclose or suggest:

assigning protection attributes indicating a memory range of a shared memory for each respective atomic sub-task of the one or

more atomic sub-tasks such that each respective sub-task is executed by one of a plurality of processors which inherits access rights to the shared memory indicated by the protection attributes corresponding to the respective atomic sub-task

as required by claim 1 and similarly by claims 8 and 16.

Koning, which is used by the Examiner to teach scheduling of one or more atomic sub-tasks into a central task queue according to one or both of temporal and priority considerations, discloses a control mechanism 614 that may be used to control access to resources that requires mutually exclusive access by task (e.g., portions of the memory space 601, and resources in the secondary storage system 602.) The control mechanism 614 may include functions to create, manage and track mutual exclusion semaphores. The control mechanism may also include functions allowing tasks to take and release mutual exclusion semaphores. (See Koning at paragraph [0074].) Koning further discloses that the memory space may be divided into a system memory space 702 generally accessible by the operating system and a user memory space 704 that may be accessed by user tasks. The system memory space may include memory space for the operating system executable code 706 and may also include space for operating system queues, including a ready queue 708 and event or time-out queue 712. Koning, however, is silent regarding, for example, a plurality of processors and, furthermore, that each respective sub-task is "executed by one of a plurality of processors which inherits access rights to the shared memory indicated by the protection attributes corresponding to the respective atomic sub-task," (emphasis added), as required by claim 1 and similarly by claims 8 and 16.

Accordingly, it is submitted that claims 2-7, 10-11 and 16-20 patentably distinguish over Kato in view of Koning for at least the above-mentioned reasons.

**Conclusion**

In view of the claim amendments and remarks, Applicants submit the application is in condition for allowance, which action is respectfully requested.

Respectfully submitted,

Lawrence E. Ashery, Reg. No. 34,515  
Eric Berkowitz, Reg. No. 44,030  
Attorneys for Applicants

LEA/EB/snp

Dated: January 9, 2008

P.O. Box 980  
Valley Forge, PA 19482  
(610) 407-0700

The Director is hereby authorized to charge or credit Deposit Account No. 18-0350 for any additional fees, or any underpayment or credit for overpayment in connection herewith.

I hereby certify that this correspondence is being electronically transmitted to: Commissioner for Patents on:

January 9, 2008

SNP\_H:\WRPORTBL\RP\SNPEREZ\234286\_1.DOC